Chart Plot

A web application that plots a line chart based on some manually input data.

Compatibility

- Google Chrome
- Microsoft Edge
- Mozilla Firefox

Technologies

- React
- Chartjs
- Sweet Alert 2
- Dirty-json
- Lodash
- Cypress

How I thought on this project?

When I finished the reading of the challenge, I thought "what a complicated project", because I have never done anything like this project, never used charts, and I'm saying that not only in programming, but in my life too, I only had done simple column charts on excel.

But I decided to start the challenge, and separate the project in parts, and only move on when the part before was finished:

1. Do the header with my name
2. Do a textarea field looking like a command line
3. Study about charts with javascript (I found chartjs/react-chartjs-2 and decided to use)
4. Implement the line chart with static data.
    1. I created a variable with different values to understand how the library worked.
    2. After that, I started to change for similar datas like the ones i would use.
    3. Finally, I changed the data to the same as I created to put on the textarea field.
5. Then, I started to think about the **data manipulation** and decided to organize with functions.
    1. On my **App.jsx** I created a button to Generate chart with a **onClick** event, calling the function **createNewChart**, this function start the process checking if has any data on textarea, showing a message with sweetalert if don't.
    2. Having any data, we enter on the formater function, who is called by **dataFormater** component, on utils.
       At this component, I followed this steps:
        - Turn each row into an array

- Try to turn the array in an object, if fail, show a error message
- Separate 'start, stop, span and data'
- Do a simple validation if textarea contains 'start, stop, spam and data' datas, returning error message if don't.
- Start using the locale functions (I described what each function do inside the code)

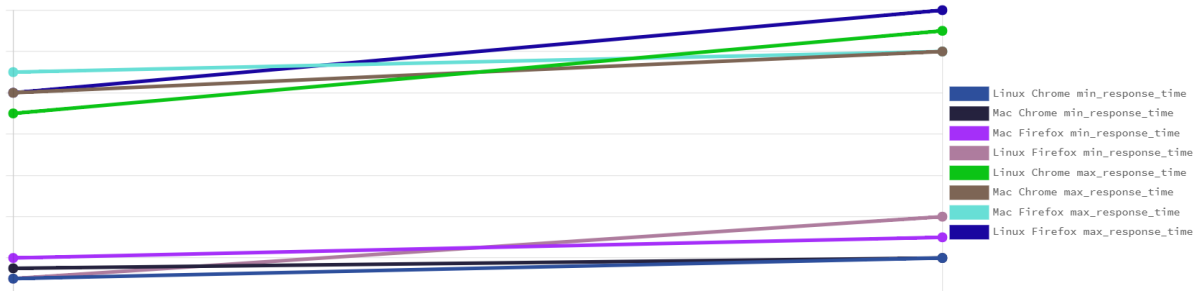
6. I created a state to put the data and the label, and inserted into **LineChart** component as props.
7. Finishing the chart part, I styled the page with a simple css archive, I don't thought necessary to use module, sass or other css technology this project.
8. After all that, I started to study about tests and I found cypress.js.
   I had never used tests on my applications, so this is the first one.
   I readed parts of the documentation and implement 4 tests:
   1. Checking if the name is showing correctly
   2. Checking if the textarea is updating the value
   3. Checking if the aplication is running when insert right format of text.
   4. Checking if the aplication is giving error when insert invalid format of text.

I think I need some practice professional experience using tests to see how it really works and what is the best way to implement, but the tests are done.

Data example to use on application:

```
{type: 'start', timestamp: 1519862400000, select: ['min_response_time', 'max_response_time'], group: ['os', 'browser']},
{type: 'span', timestamp: 1519862400000, begin: 1519862400000, end: 1519862460000},
{type: 'data', timestamp: 1519862400000, os: 'linux', browser: 'chrome', min_response_time: 0.1, max_response_time: 1.3},
{type: 'data', timestamp: 1519862400000, os: 'mac', browser: 'chrome', min_response_time: 0.2, max_response_time: 1.2},
{type: 'data', timestamp: 1519862400000, os: 'mac', browser: 'firefox', min_response_time: 0.3, max_response_time: 1.2},
{type: 'data', timestamp: 1519862400000, os: 'linux', browser: 'firefox', min_response_time: 0.1, max_response_time: 1.0},
{type: 'data', timestamp: 1519862460000, os: 'linux', browser: 'chrome', min_response_time: 0.2, max_response_time: 0.9},
{type: 'data', timestamp: 1519862460000, os: 'mac', browser: 'chrome', min_response_time: 0.15, max_response_time: 1.0},
{type: 'data', timestamp: 1519862460000, os: 'mac', browser: 'firefox', min_response_time: 0.2, max_response_time: 1.1},
{type: 'data', timestamp: 1519862460000, os: 'linux', browser: 'firefox', min_response_time: 0.4, max_response_time: 1.4},
{type: 'stop', timestamp: 1519862460000}
```

How Chart will looks like:



Matheus Rodrigues Felizardo

Linkedin: https://www.linkedin.com/in/matheus-felizardo/

Github: https://github.com/MatheusFelizardo